UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/563,409	10/26/2006	Paolo Buscema	59836-100 (09SKF106)	4575

39232 7590 07/30/2009

Themis Law
7660 Fay Ave Ste H535
La Jolla, CA 92037

EXAMINER	
HOANG, SON T	

ART UNIT	PAPER NUMBER
2165	

MAIL DATE	DELIVERY MODE
07/30/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/563,409	BUSCEMA, PAOLO	
	Examiner	Art Unit	
	SON T. HOANG	2165	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *03* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *19 May 2009*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *72-142* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *72-142* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *30 December 2005* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Response to Amendment

1. This communication is in response to the amendment filed on May 19, 2009.

Claims 72, 76, 82, 99, 107-108, 112, 118, 131, and **135** are amended.

Claims 72-142 are pending.

Response to Arguments

2. Objection to the abstract of the disclosure is maintained since no argument is presented and no correction appears to be received by the Office.

3. Objections to **claims 82, 99, 108,** and **118** are withdrawn in view of Applicant's amendment.

4. 35 U.S.C. 112, second paragraph, rejections of **claims 76, 107, 112-113,** and **116** are withdrawn in view of Applicant's amendment.

5. 35 U.S.C. 101 rejections of **claims 72-116** are withdrawn in view of Applicant's amendment.

Applicant's arguments towards **independent claims 72, 82, 99, 108, 117,** and **131** have been fully considered but are moot in view of the new ground of rejections presented hereon.

Claim Objections

6. **Claims 72**, **82**, **99**, **108**, **117**, and **131** are objected to because of the following informalities: missing antecedent basis for the limitation of "*wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables*." Appropriate correction is required.

Claim Rejections - 35 USC § 103

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

8. **Claims 72-93**, **96-97**, and **117-142** are rejected under 35 U.S.C. 103(a) as being unpatentable over Agrafiotis et al. (*Pub. No. US 2002/0091655, published on July 11, 2002; hereinafter Agrafiotis*) in view of Shmulevich et al. (*Pub. No. US 2003/0225718, filed on January 30, 2003; hereinafter Shmulevich*).

Regarding **claim 72**, Agrafiotis clearly shows and discloses a computerized method for projecting information data from a multidimensional space onto a space having lesser dimensions ([0026]), comprising:

receiving a database of N-dimensional data in the form of records having a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

defining a metric function for calculating a distance between each record in the database (*distant function d()*, [0094]);

calculating a matrix of distances between each record in the database using said metric function (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

defining a N-1 dimensional space in which each record is defined by N-1 coordinates (*for a linear from n to m dimensions, a simple 3-layer network with n input and m output units can be employed. The network is trained to reproduce the input-output coordinates produced by the iterative algorithm, and thus encodes the mapping in its synaptic parameters in a compact, analytical manner*, [0090]);

calculating the N-1 coordinates of each record in the N-1 dimensional space using an evolutionary algorithm (*coordinates of a plurality of objects on the m-dimensional nonlinear map are determined by the algorithm described in* [0050]-[0053]);

defining a best projection of the records onto the N-1 dimensional space as a projection in which a distance matrix of the records in the N-1 dimensional space one of best fits or has minimum differences with the distance matrix of the records calculated in the N-dimensional space (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The*

system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]), and

outputting said best projection of the records to a user (*Input point 1005 is projected once again, this time by the nearest local network 1021 to produce the final image 1030 on the display map*, [0117]),

wherein said receiving, defining, calculating and outputting are performed by a data processor ([0119]).

Agrafiotis does not explicitly disclose the higher dimension is N while the lesser dimension N-1, and wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables.

However, Shmulevich discloses:

the higher dimension is N while the lesser dimension N-1 (*The physical meaning behind the partial derivative of a Boolean function with respect to the ith variable is that, defined on the n-1 dimensional projection of the n-cube, it acts as an indicator of whether or not the function differs along the ith dimension*, [0116]), and

wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In*

the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software, [0124]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Shmulevich with the teachings of Agrafiotis for the purpose of modeling of complex systems, which include, but are not limited to gene regulatory networks, biological systems by utilizing methods that model the potential effect of individual genes on the global dynamical network behavior, both from the view of random gene mutation as well as intervention in order to elicit desired network behavior ([Abstract] of Shmulevich).

Regarding **claim 73**, Agrafiotis further discloses the database of N-dimensional data already contains data as to distances between the records (*Figure 1 illustrates possibilities for a single hypothetical pairwise relationship and distances of corresponding objects on a nonlinear map*, [0028]).

Regarding **claim 74**, Agrafiotis further discloses said evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space*, [0045]).

Regarding **claim 75**, Agrafiotis further discloses encoding each individual record or variable as a point having coordinates X and Y (*reducing the dimensionality of high-dimensional data in a way that preserves the original relationships of the data objects, and 2) producing Cartesian coordinate vectors from data supplied directly in the form of similarities or proximities*, [0012]);

defining a set of different X and Y coordinates for each point forming a first population of projections solution onto the lesser dimensional space (*for a nonlinear projection from n to m dimensions, a standard 3-layer neural network with n input and m output units is used. Each n-dimensional object is presented to the input layer, and its coordinates on the m-dimensional nonlinear map are obtained by the respective units in the output layer*, [0021]);

calculating a fitness score for each of the projections of the first population by using as a fitness function the matrix of distances of the single points in the original N-dimensional space (*pairwise relationship and the current distance of the corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship*, [0075]);

subjecting the population of projections to combination according to certain combinatorial rules to produce a first generation population of projections which comprises X and Y coordinates for the points which are a combination of the coordinates provided in two projections of the parent generation (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown*

in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]); and

calculating the fitness score of the projections of the first generation and forming a new generation basing on said first generation (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 76**, Shmulevich further discloses wherein the genetic algorithm is the Genetic Doping Algorithm (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and*

evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software, [0124]).

Regarding **claim 77**, Agrafiotis further discloses a hidden point is defined which corresponds to a hidden record or a to a hidden variable whose existence is only guessed at, said hidden point being added to the parent population by giving it position coordinates X_{hi} and Y_{hi} in the projection (*multi-layer feed-forward networks trained to reproduce their inputs as desired outputs. They consist of an input and an output layer containing as many neurons as the number of input dimensions, and a series of hidden layers having a smaller number of units. In the first part of the network, each sample is reorganized, mixed, and compressed into a compact representation encoded by the middle layer. This representation is then decompressed by the second part of the network to reproduce the original input*, [0011]).

Regarding **claim 78**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel with and without the hidden point and wherein best fit projections obtained from said two parallel calculations are compared (*a phase of global refinement is initially carried out, after which, the resulting nonlinear map is partitioned into a regular grid. The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global*

refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance, [0086]).

Regarding **claim 79**, Agrafiotis further discloses:

providing a database comprising at least two records, each record having a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

elaborating the database alternatively or in parallel: in a first manner in which the records are considered as being points and the variables as the coordinates of the points; and in a second manner in which the variables are considered as being points and the records as the coordinates of the points (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]).

Regarding **claim 80**, Agrafiotis further discloses operating on the database in a pre-processing or post-processing phase (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]).

Regarding **claim 81**, Agrafiotis further discloses the database is processed in a preventive stage by means of a Self Organising Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated*

relationships at a time, [0047]), and wherein the clusters formed by this algorithm in the different units are projected to said lesser dimensional space (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step 325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335*, [0097]).

Regarding **claim 82**, Agrafiotis clearly shows and discloses a computerized method for the cognitive analysis of multidimensional information data ([0026]), comprising:

receiving a database with a certain number of records, each record comprising a certain number of variables and being relative to a N-dimensional space (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

projecting the database onto a space having a lesser number of dimensions relative to the N-dimensional space, considering the records as points and the variables as coordinates, or the variables as points and the records as coordinates (*For a nonlinear projection from n to m dimensions, a simple 3-layer network with n input and m output units can be employed. The network is trained to reproduce the input/output coordinates produced by the iterative algorithm, and thus encodes the mapping in its synaptic parameters in a compact, analytical manner*, [0090]);

carrying out said projection using an algorithm for projecting information data belonging to a multidimensional space into a space having lesser dimensions by (*The process described herein uses the iterative nonlinear mapping algorithm described in*

Section II to multidimensionally scale a small random sample of a set of input patterns of dimensionality n, and then "learns" the underlying nonlinear transform using an artificial neural network, [0090]):

calculating a matrix of distances between each point defined by a record or a variable of the database using a metric function (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

defining a N-1 dimensional space in which each point represented by a record or a variable is defined by N-1 coordinates (*for a linear from n to m dimensions, a simple 3-layer network with n input and m output units can be employed. The network is trained to reproduce the input-output coordinates produced by the iterative algorithm, and thus encodes the mapping in its synaptic parameters in a compact, analytical manner*, [0090]);

calculating the N-1 coordinates of each point in the N-1 dimensional space using an evolutionary algorithm (*coordinates of a plurality of objects on the m-dimensional nonlinear map are determined by the algorithm described in* [0050]-[0053]); and

generating a best projection in which the distance matrix of the points in the N-1 dimensional space best fits or has minimum differences with the distance matrix of the points calculated in the N-dimensional space as the best projection of said points onto the N-1 dimensional space (*Note that new patterns in R^n that not in the original input set*

can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]); and

outputting said best projection to a user (*Input point 1005 is projected once again, this time by the nearest local network 1021 to produce the final image 1030 on the display map*, [0117]),

wherein said receiving, projecting, and carrying out are performed by a data processor ([0119]).

Agrafiotis does not explicitly disclose the higher dimension is N while the lesser dimension N-1, and wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables.

However, Shmulevich discloses:

the higher dimension is N while the lesser dimension N-1 (*The physical meaning behind the partial derivative of a Boolean function with respect to the ith variable is that,*

defined on the n-1 dimensional projection of the n-cube, it acts as an indicator of whether or not the function differs along the ith dimension, [0116]), and

wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software*, [0124]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Shmulevich with the teachings of Agrafiotis for the purpose of modeling of complex systems, which include, but are not limited to gene regulatory networks, biological systems by utilizing methods that model the potential effect of individual genes on the global dynamical network behavior, both from the view of random gene mutation as well as intervention in order to elicit desired network behavior ([Abstract] of Shmulevich).

Regarding **claim 83**, Agrafiotis further discloses the database already contains distances between the records (*Figure 1 illustrates possibilities for a single hypothetical pairwise relationship and distances of corresponding objects on a nonlinear map*, [0028]).

Regarding **claim 84**, Agrafiotis further discloses the evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset*

refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space, [0045]).

Regarding **claim 85**, Agrafiotis further discloses said analysis:

encodes each individual record or variable as a point having coordinates X and Y (*reducing the dimensionality of high-dimensional data in a way that preserves the original relationships of the data objects, and 2) producing Cartesian coordinate vectors from data supplied directly in the form of similarities or proximities*, [0012]);

defines a set of different X and Y coordinates for each point forming a first population of projections solution onto the less dimensional space, usually a two or three dimensional space (*for a nonlinear projection from n to m dimensions, a standard 3-layer neural network with n input and m output units is used. Each n-dimensional object is presented to the input layer, and its coordinates on the m-dimensional nonlinear map are obtained by the respective units in the output layer*, [0021]);

calculates a fitness score for each of the projections of this first population by using the matrix of distances of the single points in the original N-dimensional space as a fitness function (*pairwise relationship and the current distance of the corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship*, [0075]);

combines the population of projections according to certain combinatorial rules to produce a first generation population of projections which comprises X and Y coordinates for the points which are a combination of the coordinates provided in two projections of the parent generation (*Note that new patterns in R^1 that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]);

calculates a fitness score of the projections of the first generation; and forms a new generation based on the first generation (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 86**, Shmulevich further discloses the evolutionary algorithm is the GenD algorithm (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software*, [0124]).

Regarding **claim 87**, Agrafiotis further discloses a hidden point represented by a hidden record or a hidden variable can be defined, which corresponds to a hidden point on the map and whose existence is only guessed at, said hidden point being added to the parent population by giving it position coordinates X_{hi} and Y_{hi} in the projection (*multi-layer feed-forward networks trained to reproduce their inputs as desired outputs. They consist of an input and an output layer containing as many neurons as the number of input dimensions, and a series of hidden layers having a smaller number of units. In the first part of the network, each sample is reorganized, mixed, and compressed into a compact representation encoded by the middle layer. This representation is then decompressed by the second part of the network to reproduce the original input*, [0011]).

Regarding **claim 88**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel, with and without the hidden point, and best fit projections obtained by said two parallel calculations are compared (*a phase of global refinement is initially carried out, after which, the resulting nonlinear map is*

partitioned into a regular grid. The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance, [0086]).

Regarding **claim 89**, Agrafiotis further discloses:

providing a database comprising a certain number of records each one having a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

elaborating the database alternatively or in parallel according to: a first manner by which the records are considered as being points and the variables as being the coordinates of the points; and a second manner by which the variables are considered as being points and the records are the coordinates (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]).

Regarding **claim 90**, Agrafiotis further discloses using a different algorithm to operate on the database in a pre-processing or a post-processing phase (*The method uses a self-organizing principle to iteratively refine an initial (random or partially*

ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time, [0047]).

Regarding **claim 91**, Agrafiotis further discloses the database is processed in a preventive stage by means of a Self Organising Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]), and wherein the clusters formed by said SOP algorithm in the different units are projected onto said lesser dimensional space (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step 325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335*, [0097]).

Regarding **claim 92**, Agrafiotis further discloses the distance of the points on the map onto which the database has been projected is used as a measure of similarity of the records or variables related to said points (*The general algorithm can also be applied when the pairwise similarity matrix is incomplete, i.e. when some of the pairwise similarities are unknown, when some of the pairwise similarities are uncertain or corrupt, or both of the above*, [0066]).

Regarding **claim 93**, Agrafiotis further discloses:

a database is provided comprising a certain number of records, each record related to a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

complementary variables to the variables originally provided are added to the database (*the basic algorithm does not distinguish weak from strong relationships (short-range and long-range distances, respectively). One method to ensure that strong relationships are preserved more faithfully than weak relationships is to weight the coordinate update in Equation 5 (or, equivalently, the learning rate λ in Equations 7 and 8) by a scaling factor that is inversely proportional to the strength (magnitude) of the relationship*, [0085]);

the integrated database is projected onto a lesser dimensional space, particularly on a two or three dimensional space (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]); and

the distance between each variable and its complementary variable in said mapping is used as a measure of the relevance of said variable in the database (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]).

Regarding **claim 96**, Agrafiotis further discloses implemented as a set of computer executable instructions saved on a removable computer readable storage medium (*removable storage unit storing computer software*, [0120]).

Regarding **claim 97**, Agrafiotis further discloses implemented as a set of computer executable instructions saved on a removable computer readable storage medium (*removable storage unit storing computer software*, [0120]).

Regarding **claim 117**, Agrafiotis further discloses an apparatus having artificial intelligence provided with a processing unit, said processing unit connected to a data memory and to a program memory ([0120]), wherein:

the processing unit is further connected to one or more different sensors for detecting or measuring different physical and/or chemical conditions or effects or processes characterizing or occurring in an environment (*Figure 11 shows the computer system 1000 includes one or more processors, such as processor 1104. The processor 1104 is connected to a communication infrastructure 1106 (e.g., a bus or network)*, [0119]. *See further* [0011] *for neurons acting as sensors*);

the processing unit is further connected to data input means by means of a service person or a data input line from other data collecting apparati; the processing unit is also connected to means for carrying out mechanical, physical or chemical actions, such as actuators or the like (*Secondary memory 1110 can also include other similar means for allowing computer programs or input data to be loaded into computer system 1100. Such means may include, for example, a removable storage unit 1122 and an interface 1120. Examples of such may include a program cartridge and cartridge interface (such as that found in video game devices), a removable memory chip (such as an EPROM, or PROM) and associated socket, and other removable storage units 1122 and interfaces 1120 which allow software and data to be transferred from the removable storage unit 1122 to computer system 1100*, [0121]);

in the program memory a program executable by the processing unit being loaded which program has a routine for driving the sensors and saving in a uniquely recognizable way each datum collected by the sensors and/or for saving data input by a service person or by other apparati, driver for activating or deactivating the means for carrying out mechanical, physical or chemical actions, such as actuators or the like (*Computer programs (also called computer control logic) are stored in main memory 1108 and/or secondary memory 1110. Computer programs may also be received via communications interface 1124. Such computer programs, when executed, enable the computer system 1100 to perform the features of the present invention as discussed herein. In particular, the computer programs, when executed, enable the processor*

1104 to perform the features of the present invention. Accordingly, such computer programs represent controllers of the computer system 1100, [0124]);

the program stored in the program memory further comprising means for evaluating the data collected by the sensors and/or the data inputted by a service person or by other apparati (*Computer system 1100 may also include a communications interface 1124. Communications interface 1124 allows software and data to be transferred between computer system 1100 and external devices*, 0122]);

wherein the program includes a subroutine for executing an algorithm implementing the method of claim 72 on the collected and/or inputted data (*mapping of massive data sets from a higher dimensional space to a lower dimensional space. Moreover, the method allows the mapping of new input patterns as they become available, without the need to reconstruct an entire map*, [0026]).

Regarding **claim 118**, Agrafiotis further discloses the program is provided with a subroutine for carrying out a cognitive analysis of multidimensional data by treating the said data with a method for projecting information data belonging to a multidimensional space into a space having less dimensions ([0026]) comprising the following steps:

providing a database of N-dimensional data in the form of records having a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

defining a metric function for calculating a distance between each record of the database (*distant function d()*, [0094]);

calculating a matrix of distances between each record of the database by means of the metric function defined at the previous step (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

defining a N-1 dimensional space in which each record is defined by N-1 coordinates (*for a linear from n to m dimensions, a simple 3-layer network with n input and m output units can be employed. The network is trained to reproduce the input-output coordinates produced by the iterative algorithm, and thus encodes the mapping in its synaptic parameters in a compact, analytical manner*, [0090]);

calculating the N-1 coordinates of each record in the N-1 dimensional space by means of an evolutionary algorithm (*coordinates of a plurality of objects on the m-dimensional nonlinear map are determined by the algorithm described in* [0050]-[0053]);

defining as the best projection of the records onto the N-1 dimensional space the projection in which the distance matrix of the records in the N-1 dimensional space best fits or has minimum differences with the distance matrix of the records calculated in the n-dimensional space (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system*

includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]).

Agrafiotis does not explicitly disclose the higher dimension is N while the lesser dimension N-1.

However, Stolte discloses the higher dimension is N while the lesser dimension N-1 (*Projection (Time, Product, Location → Time, Location) reduces the dimensionality of an n-dimensional data cube to (n-1) by aggregating across a dimension. For example, in Figure 15, the first projection summarizes across "Location", reducing the 3-dimensional cube to a 2-dimensional cube*, [0189]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Stolte with the teachings of Agrafiotis for the purpose of providing an interactive visual exploration tool that facilitates exploratory analysis of databases having a hierarchical structure ([0038] of Stolte).

Regarding **claim 119**, Agrafiotis further discloses said database already contains the distances between the records (*Figure 1 illustrates possibilities for a single hypothetical pairwise relationship and distances of corresponding objects on a nonlinear map*, [0028]).

Regarding **claim 120**, Agrafiotis further discloses said evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space*, [0045]).

Regarding **claim 121**, Agrafiotis further discloses said subroutine provides for:

encoding each individual record or variable represented by a point having coordinate X and Y (*reducing the dimensionality of high-dimensional data in a way that preserves the original relationships of the data objects, and 2) producing Cartesian coordinate vectors from data supplied directly in the form of similarities or proximities*, [0012]);

defining a set of different X and Y coordinates for each point forming a first population of projections solution onto the less dimensional space, usually a two or three dimensional space (*for a nonlinear projection from n to m dimensions, a standard 3-layer neural network with n input and m output units is used. Each n-dimensional object is presented to the input layer, and its coordinates on the m-dimensional nonlinear map are obtained by the respective units in the output layer*, [0021]);

calculating the fitness score for each of the projections of this first population by using as the fitness function the matrix of distances of the single points in the originally N dimensional space (*pairwise relationship and the current distance of the*

corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship, [0075]);

subjecting the population of projections to combination according to certain combination rules thus producing a first generation population of projections which comprises X and Y coordinates for the points which are a combination of the coordinates provided in two projections of the parent generation (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]); and

calculating the fitness score of the projections of the first generation and forming again a new generation basing on the first generation (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This*

process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance, [0086]).

Regarding **claim 122**, Shmulevich further discloses the genetic algorithm is the GenD algorithm (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software*, [0124])..

Regarding **claim 123**, Agrafiotis further discloses a hidden point can be defined which corresponds to a hidden record or a to a hidden variable whose existence is only speculative, and wherein said hidden point is added in the parent population by giving it positional coordinates X_{hi} and Y_{hi} in the projection (*multi-layer feed-forward networks trained to reproduce their inputs as desired outputs. They consist of an input and an output layer containing as many neurons as the number of input dimensions, and a series of hidden layers having a smaller number of units. In the first part of the network, each sample is reorganized, mixed, and compressed into a compact representation encoded by the middle layer. This representation is then decompressed by the second part of the network to reproduce the original input*, [0011]).

Regarding **claim 124**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel with the hidden point and without the hidden point and the best fit projections obtained by the two parallel calculations are compared (*a phase of global refinement is initially carried out, after which, the resulting nonlinear map is partitioned into a regular grid. The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 125**, Agrafiotis further discloses:

providing a database comprising a certain number of records each one characterized by a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

elaborating the database alternatively or in parallel according to: a first manner in which the records are considered as being points and the variables as being the coordinates of the points; and a second manner in which the variables are considered as being points and the records are the coordinates (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points* X_{i-k}

and the set of reference points (both of which were extracted from the input space R^n is calculated, [0096]-[0097]).

Regarding **claim 126**, Agrafiotis further discloses a different treatment of the data as a pre or post processing phase (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]).

Regarding **claim 127**, Agrafiotis further discloses characterized in that the data is processed in a preventive step by means of a Self Organizing Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]), the clusters formed by said SOM algorithm in the different units then being projected onto said space of lesser dimension (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step 325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335*, [0097]).

Regarding **claim 128**, Agrafiotis further discloses a relationship between the collected data records of the collected data is determined by means of the distance of each data record from the other data records and wherein said distance is used as a relevance weight for each data record in determining the activation or deactivation of one or more of the means for carrying out mechanical, physical and/or chemical actions

(*the basic algorithm does not distinguish weak from strong relationships (short-range and long-range distances, respectively). One method to ensure that strong relationships are preserved more faithfully than weak relationships is to weight the coordinate update in Equation 5 (or, equivalently, the learning rate λ in Equations 7 and 8) by a scaling factor that is inversely proportional to the strength (magnitude) of the relationship*, [0085]).

Regarding **claim 129**, Agrafiotis further discloses a maximum distance for each data record in determining the activation or record from the other is set for discriminating the data records to be used in determining the activation or deactivation of one or more of the means for carrying out said mechanical, physical and or chemical actions (*If d_{ij} is larger than r_{max}, the coordinates of the objects are updated using r_{max} as the target distance (Equation 9)*, [0071]).

Regarding **claim 130**, Agrafiotis further discloses the clustering or distance of the data records on the map onto which the database has been projected is used as a measure of similarity of the data records or of the variables related to the said data records (*the algorithm attempts to match the upper bound if the current distance between the objects is greater than the upper bound, or the lower bound if the current distance between the objects is lower than the lower bound. If the distance between the objects lies within the upper and lower bounds, no correction is made*, [0073]).

Regarding **claim 131**, Agrafiotis clearly shows and discloses an apparatus having artificial intelligence containing a simulation of intuitive behaviour ([0120]), comprising:

a processing unit connected to each of a data memory and a program memory, wherein the processing unit is further connected to one or more sensors for detecting or measuring different physical and/or chemical conditions or effects or processes characterising or occurring in the environment (*Figure 11 shows the computer system 1000 includes one or more processors, such as processor 1104. The processor 1104 is connected to a communication infrastructure 1106 (e.g., a bus or network)*, [0119]. *See further* [0011] *for neurons acting as sensors*);

wherein the processing unit is further connected to data input means by means of a service person or a data input line from other data collecting apparati, wherein the processing unit is further connected to means for carrying out mechanical, physical or chemical actions, such as actuators or the like (*Secondary memory 1110 can also include other similar means for allowing computer programs or input data to be loaded into computer system 1100. Such means may include, for example, a removable storage unit 1122 and an interface 1120. Examples of such may include a program cartridge and cartridge interface (such as that found in video game devices), a removable memory chip (such as an EPROM, or PROM) and associated socket, and other removable storage units 1122 and interfaces 1120 which allow software and data to be transferred from the removable storage unit 1122 to computer system 1100*, [0121]);

wherein the program memory stores a program executable by the processing unit, said program including a routine for driving the sensors and saving in a uniquely recognizable way each datum collected by the sensors and/or for saving data input by a service person or by other apparati, driver for activating or deactivating the means for carrying out mechanical, physical or chemical actions, such as actuators or the like (*Computer programs (also called computer control logic) are stored in main memory 1108 and/or secondary memory 1110. Computer programs may also be received via communications interface 1124. Such computer programs, when executed, enable the computer system 1100 to perform the features of the present invention as discussed herein. In particular, the computer programs, when executed, enable the processor 1104 to perform the features of the present invention. Accordingly, such computer programs represent controllers of the computer system 1100*, [0124]);

wherein the program stored in the program memory further comprises means for evaluating data collected by the sensors and/or inputted by a service person or by other apparati (*Computer system 1100 may also include a communications interface 1124. Communications interface 1124 allows software and data to be transferred between computer system 1100 and external devices*, 0122]); and

wherein in operation the apparatus:

generates from the collected and/or inputted data a database with a certain number of records each one comprising a certain number of variables and which are

relative to a N-dimensional space (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

projects the database considering the record as points and the variables as coordinates or the variables as points and the records as coordinates onto a space having a reduced number of dimension relatively to the space N dimensional space (*mapping of massive data sets from a higher dimensional space to a lower dimensional space*, [0026]), said projection being carried out by means of an algorithm for projecting information data belonging to a multidimensional space into a space having lesser dimensions comprising: calculating a matrix of distances between each point defined by a record or a variable of the database by means of a metric function (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

defining a N-1 dimensional space in which each point represented by a record or a variable is defined by N-1 coordinates (*for a linear from n to m dimensions, a simple 3-layer network with n input and m output units can be employed. The network is trained to reproduce the input-output coordinates produced by the iterative algorithm, and thus encodes the mapping in its synaptic parameters in a compact, analytical manner*, [0090]);

calculating the N-1 coordinates of each point in the N-1 dimensional space by means of an evolutionary algorithm (*coordinates of a plurality of objects on the m-*

dimensional nonlinear map are determined by the algorithm described in [0050]-[0053]); and

defining as the best projection of the points onto the N-1 dimensional space the projection in which the distance matrix of the points in the n-1 dimensional space best fits or has minimum differences with the distance matrix of the points calculated in the N-dimensional space (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]).

Agrafiotis does not explicitly disclose the higher dimension is N while the lesser dimension N-1.

However, Stolte discloses the higher dimension is N while the lesser dimension N-1 (*Projection (Time, Product, Location → Time, Location) reduces the dimensionality of an n-dimensional data cube to (n-1) by aggregating across a dimension. For*

example, in Figure 15, the first projection summarizes across "Location", reducing the 3-dimensional cube to a 2-dimensional cube, [0189]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Stolte with the teachings of Agrafiotis for the purpose of providing an interactive visual exploration tool that facilitates exploratory analysis of databases having a hierarchical structure ([0038] of Stolte).

Regarding **claim 132**, Agrafiotis further discloses said database already contains distances between the records (*Figure 1 illustrates possibilities for a single hypothetical pairwise relationship and distances of corresponding objects on a nonlinear map*, [0028]).

Regarding **claim 133**, Agrafiotis further discloses the evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space*, [0045]).

Regarding **claim 134**, Agrafiotis further discloses encoding each individual record or variable represented by a point having coordinates X and Y;

defining a set of different X and Y coordinates for each point forming a first population of projections solution onto the lesser dimensional space, usually a two or

three dimensional space (*reducing the dimensionality of high-dimensional data in a way that preserves the original relationships of the data objects, and 2) producing Cartesian coordinate vectors from data supplied directly in the form of similarities or proximities*, [0012]);

calculating a fitness score for each of the projections of said first population by using as a fitness function the matrix of distances between the single points in the original N dimensional space (*pairwise relationship and the current distance of the corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship*, [0075]);

combining the population of projections according to certain combinatorial rules and producing a first generation population of projections which comprises X and Y coordinates for the points which are a combination of the coordinates provided in two projections of the parent generation (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point*

nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]); and

calculating the fitness score of the projections of the first generation and forming a new generation basing on the first generation (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 135**, Shmulevich further discloses the evolutionary algorithm is the Genetic Doping Algorithm (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software*, [0124]).

Regarding **claim 136**, Agrafiotis further discloses a hidden point represented by a hidden record or a hidden variable can be defined, which corresponds to a hidden point on the map whose existence is only speculated, and wherein said hidden point is added to the parent population by giving it positional coordinates X_{hi} and Y_{hi} in the

projection (*multi-layer feed-forward networks trained to reproduce their inputs as desired outputs. They consist of an input and an output layer containing as many neurons as the number of input dimensions, and a series of hidden layers having a smaller number of units. In the first part of the network, each sample is reorganized, mixed, and compressed into a compact representation encoded by the middle layer. This representation is then decompressed by the second part of the network to reproduce the original input*, [0011]).

Regarding **claim 137**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel with and without the hidden point and wherein the best fit projections obtained by the two parallel calculations are compared (*a phase of global refinement is initially carried out, after which, the resulting nonlinear map is partitioned into a regular grid. The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 138**, Agrafiotis further discloses:

providing a database comprising a certain number of records each one characterised by a certain number of variables (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

elaborating the database alternatively or in parallel according to: a first manner in which the records are considered as being points and the variables as being the coordinates of the points; and a second manner in which the variables are considered as being points and the records are the coordinates (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]).

Regarding **claim 139**, Agrafiotis further discloses treating the database as a pre-processing or post-processing phase (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]).

Regarding **claim 140**, Agrafiotis further discloses the database is processed in a preventive stage by means of a Self Organising Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]), the clusters formed by said SOP algorithm in the different units being projected onto said space of lesser dimension (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step*

325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335, [0097]).

Regarding **claim 141**, Agrafiotis further discloses the clustering or distance of the points on the map onto which the database has been projected is used as a measure of similarity of the records or of the variables related to the said point (*the algorithm attempts to match the upper bound if the current distance between the objects is greater than the upper bound, or the lower bound if the current distance between the objects is lower than the lower bound. If the distance between the objects lies within the upper and lower bounds, no correction is made*, [0073]).

Regarding **claim 142**, Agrafiotis further discloses:

a database comprising a certain number of records, each record being related to a certain number of variables, is provided (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

variables complementary to the variables originally provided are added to said database (*the basic algorithm does not distinguish weak from strong relationships (short-range and long-range distances, respectively). One method to ensure that strong relationships are preserved more faithfully than weak relationships is to weight the coordinate update in Equation 5 (or, equivalently, the learning rate λ in Equations 7 and 8) by a scaling factor that is inversely proportional to the strength (magnitude) of the relationship*, [0085]) and the resulting integrated database is projected onto a space of

lesser dimension, particularly onto a two or three dimensional space (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]); and

the distance in the map between each variable and its complementary variable is used as a measure of the relevance of said variable in the database (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]).

9. **Claims 94-95** are rejected under 35 U.S.C. 103(a) as being unpatentable over Agrafiotis et al. (*Pub. No. US 2002/0091655, published on July 11, 2002; hereinafter Agrafiotis*) in view of Shmulevich et al. (*Pub. No. US 2003/0225718, filed on January 30, 2003; hereinafter Shmulevich*), and further in view of Granger (*Pat. No. US 6,463,321, published on September 6, 2007*).

Regarding **claim 94**, Agrafiotis, as modified by Shmulevich, does not disclose said method is used to evaluate the relevance of certain variables in determining a certain pathological status of individuals, and to define prototypes of individuals relative to the variables of the database and their probability of developing a certain disease.

However, Granger discloses said method is used to evaluate the relevance of certain variables in determining a certain pathological status of individuals, and to define prototypes of individuals relative to the variables of the database and their probability of developing a certain disease (*projections of the characterizing ERP signal vector are generated. A projection of a vector is produced when some of the coordinates of the vector are eliminated. Generally, " projection" refers to the operation of transforming high -dimensional data into lower -dimensional spaces. An example would be the viewing of three-dimensional images in a two-dimensional plane (e.g., a photograph)*, [0032]. *See* [0010] *for diagnosing the presence of a neurological disorder (such as Alzheimer's Disease, depression, or schizophrenia), otherwise assessing the neurological condition of a patient, or characterizing the results of a treatment regimen used by a patient*).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Granger with the teachings of Agrafiotis, as modified by Shmulevich, for the purpose of assessing the neurological condition of a patient, where the method obtains and processes relatively consistent, low noise, artifact-free data in a fast, inexpensive, and reliable ways ([0009] of Granger).

Regarding **claim 95**, Granger further discloses analyze the probability of an individual of having or developing Alzheimer's (*diagnosing the presence of a neurological disorder (such as Alzheimer's Disease, depression, or schizophrenia), otherwise assessing the neurological condition of a patient, or characterizing the results of a treatment regimen used by a patient*, [0010]).

10. **Claims 98-106** are rejected under 35 U.S.C. 103(a) as being unpatentable over Agrafiotis et al. (*Pub. No. US 2002/0091655, published on July 11, 2002; hereinafter Agrafiotis*) in view of Shmulevich et al. (*Pub. No. US 2003/0225718, filed on January 30, 2003; hereinafter Shmulevich*), and further in view of Schipper (*Pat. No. US 5,581,259, published on December 3, 1996*).

Regarding **claim 98**, Agrafiotis, as modified by Shmulevich, does not disclose that the mapping process is for geographical sites.

However, Schipper discloses mapping process if for geographical sites (*Figure 5 is a projection of a three-dimensional region on a two-dimensional surface showing the projected locations of three landmarks, L1, L2 and L3, that are also shown on an old map as L1', L2' and L3' in Figure 5, with the respective location coordinates (x1',y1',z1'), (x2',y2',z2') and (x3',y3',z3'*, [Column 25, Lines 37-43]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Schipper with the teachings of

Agrafiotis, as modified by Shmulevich, for the purpose of allowing reconciliation of a more accurate navigation system with a less accurate physical map or chart to provide increased accuracy for the combination ([Column 4, Line 65 → Column 5, Line 4] of Schipper).

Regarding **claim 99**, Agrafiotis clearly shows and discloses a computerized method for generating two or three dimensional maps of data records starting from a database containing relative distances between the records ([0026]), comprising:

organizing the known or measured distance values of the data records in a matrix form (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

defining a two or a three dimensional space in which the position of each record is uniquely defined by two or three coordinates (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

determining the positional coordinates of each data records in the two or three dimensional space using an evolutionary algorithm (*coordinates of a plurality of objects on the m-dimensional nonlinear map are determined by the algorithm described in* [0050]-[0053]);

determining the distances of the data records one from the other using the calculated two or three dimensional positional coordinates of said data records one from the other (*training set T is extracted from the set of input patterns n, the difference in*

distance between a random set of points X_{i+k} and the set of reference points (both of which were extracted from the input space R^n is calculated, [0096]-[0097]);

generating a matrix of distances with distance values determined according to said means; and defining as the best set of two or three dimensional coordinates of position of the totality of the data records in the two or three dimensional space, the two or three dimensional coordinates of position of the said data records for which the distance matrix determined therefrom best fits or has a minimum difference with the distance matrix of the known or measured distance values of the data records (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]).

Agrafiotis does not explicitly disclose the higher dimension is 3-dimensional while the lesser dimension 2-dimensional, and wherein in said evolutionary algorithm the

number of marriages and of mutations of individuals are adaptive self-definable internal variables.

However, Shmulevich discloses:

the higher dimension is N while the lesser dimension N-1 (*The physical meaning behind the partial derivative of a Boolean function with respect to the ith variable is that, defined on the n-1 dimensional projection of the n-cube, it acts as an indicator of whether or not the function differs along the ith dimension*, [0116]), and

wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables (*Suppose that any gene out of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software*, [0124]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Shmulevich with the teachings of Agrafiotis for the purpose of modeling of complex systems, which include, but are not limited to gene regulatory networks, biological systems by utilizing methods that model the potential effect of individual genes on the global dynamical network behavior, both from the view of random gene mutation as well as intervention in order to elicit desired network behavior ([Abstract] of Shmulevich).

Agrafiotis, as modified by Shmulevich, does not disclose that the mapping process is for geographical sites, and outputting said best set of coordinates of position to a user.

However, Schipper discloses the mapping process is for geographical sites, and outputting said best set of coordinates of position to a user (*Figure 5 is a projection of a three-dimensional region on a two-dimensional surface showing the projected locations of three landmarks, L1, L2 and L3, that are also shown on an old map as L1', L2' and L3' in Figure 5, with the respective location coordinates (x1',y1',z1'), (x2',y2',z2') and (x3',y3',z3'*, [Column 25, Lines 37-43]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Schipper with the teachings of Agrafiotis, as modified by Shmulevich, for the purpose of allowing reconciliation of a more accurate navigation system with a less accurate physical map or chart to provide increased accuracy for the combination ([Column 4, Line 65 → Column 5, Line 4] of Schipper).

Regarding **claim 100**, Agrafiotis further discloses the evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space*, [0045]).

Regarding **claim 101**, Agrafiotis further discloses:

a) for each geographical site a first and a second set of coordinates defining the position of each geographical site in the two or three dimensional space is calculated (*for a nonlinear projection from n to m dimensions, a standard 3-layer neural network with n input and m output units is used. Each n-dimensional object is presented to the input layer, and its coordinates on the m-dimensional nonlinear map are obtained by the respective units in the output layer*, [0021]);

b) a fitness score of the matrix of distances among the geographical sites determined by means of the first and second set of coordinates defining the position of each geographical site in the two or three dimensional space is calculated by using as a fitness function the matrix of the known or measured distances of the geographical sites (*pairwise relationship and the current distance of the corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship*, [0075]);

c) the first and second set of coordinates of position of each geographical site are combined for each geographical site according to predetermined combination rules, thus producing at least a new first and second set of coordinates of position for each geographical site (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is*

a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m, [0110]);

d) a fitness score of the said new first and second set of coordinates of position according to (b) is calculated (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]); and

e) said new first and second set of coordinates of position are further combined according to (c), and (c) through (e) are repeated until at least one new first or second set of positional coordinates reaches a maximum fitness score or is greater than a minimum predefined fitness score (*If d_{ij} is larger than r_{max}, the coordinates of the objects are updated using r_{max} as the target distance (Equation 9)*, [0071]).

Regarding **claim 102**, Agrafiotis further discloses for each combination of at least one first and one second set of coordinates several new sets of coordinates are obtained by the combinations of the said at least first and second set of coordinates (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]).

Regarding **claim 103**, Agrafiotis further discloses at least one hidden or hypothetical geographical site is added to the database of geographical sites of which neither the coordinates nor the distances are known and wherein a first and a second set of coordinates for said at least one geographical site are freely defined (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time. The relationship data may be complete or incomplete (i.e. some relationships between objects may not be known), exact or inexact (i.e. some*

or all relationships may be given in terms of allowed ranges or limits), symmetric or asymmetric (i.e. the relationship of object A to object B may not be the same as the relationship of B to A) and may contain systematic or stochastic errors, [0047]).

Regarding **claim 104**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel for the database with and without the hidden or hypothetical geographical site, and wherein the best fit set of positional coordinates of the totality of geographical sites obtained by said two parallel calculations are compared (*a phase of global refinement is initially carried out, after which, the resulting nonlinear map is partitioned into a regular grid. The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 105**, Agrafiotis further discloses an additional pre-processing or post-processing phase is provided (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]).

Regarding **claim 106**, Agrafiotis further discloses in a preventive phase the known distance data matrix is subjected to treatment by means of a Self Organising Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]), and wherein the clusters formed by said SOM algorithm in the different units are then projected into said lesser dimensional space (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step 325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335*, [0097]).

11. **Claims 107-116** are rejected under 35 U.S.C. 103(a) as being unpatentable over Agrafiotis et al. (*Pub. No. US 2002/0091655, published on July 11, 2002; hereinafter Agrafiotis*) in view of Shmulevich et al. (*Pub. No. US 2003/0225718, filed on January 30, 2003; hereinafter Shmulevich*), and further in view of Blaney et al. (*Pat. No. US 5,680,331, published on October 21, 1997; hereinafter Blaney*).

Regarding **claim 107**, Agrafiotis, as modified by Shmulevich, does not disclose said method is used to represent the structure of a molecule in a three dimensional or two dimensional space, by indicating only the relative distances of the atoms of the molecule.

However, Blaney discloses the structure of a molecule in a three dimensional or two dimensional space. by indicating only the relative distances of the atoms of the molecule (*Figure 1a is a representation of a protein, namely epidermal growth factor (EGF), showing its alpha carbon backbone B and the side chains TYR 13 (Tyrosine), CYS 14 (Cysteine), LEU 15 (Leucine), ASN 16 (Asparagine), and ARG 41 (Arginine), CYS 42 (Cysteine), and GLN 43 (Glutamine) of its putative active site AS. EGF has a complicated three-dimensional configuration, of which Figure 1a is a two-dimensional projection. Figure 1b is another projection of the molecule, rotated 90 degrees around the Y-axis which is shown for reference in each of these two figures*, [Column 3, Lines 26-37]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Blaney with the teachings of Agrafiotis, as modified by Shmulevich, for the purpose of identifying the putative active site of a molecule, determining the structure of the active site, and, by means of a depth-first recursive procedure, generating designs of molecules which mimic that site to a desired degree of accuracy ([Column 1, Lines 58-64] of Blaney).

Regarding **claim 108**, Agrafiotis further discloses a method for representing the structure of a database in a three dimensional or two dimensional space by indicating only the relative distances of at least part of the data records of the database relative to one another, comprising:

a) organizing known or measured distance values of the data records in a matrix form (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

b) defining a two or a three dimensional space in which the position of each data records is uniquely defined by two or three coordinates (*a set of input patterns of dimensionality n, each with corresponding coordinates*, [0090]);

c) determining the two or three coordinates of the position in the two or three dimensional space of each data record by means of an evolutionary algorithm (*coordinates of a plurality of objects on the m-dimensional nonlinear map are determined by the algorithm described in* [0050]-[0053]);

d) determining the distances of the data records one from the other by means of the calculated two or three dimensional coordinates of position of the said data records one from the other (*training set T is extracted from the set of input patterns n, the difference in distance between a random set of points X_{i-k} and the set of reference points (both of which were extracted from the input space R^n is calculated*, [0096]-[0097]);

e) generating a matrix of distances with distances determined according to (d); and f) defining as the best two or three dimensional coordinates of position of the totality of the data records in the two or three dimensional space, the two or three dimensional coordinates of position of the said data records for which the distance

matrix determined therefrom best fits, or has a minimum difference, with the distance matrix of the known or measured distance values of the data records (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]).

Agrafiotis does not explicitly disclose the higher dimension is N while the lesser dimension N-1, and wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables.

However, Shmulevich discloses:

the higher dimension is N while the lesser dimension N-1 (*The physical meaning behind the partial derivative of a Boolean function with respect to the ith variable is that, defined on the n-1 dimensional projection of the n-cube, it acts as an indicator of whether or not the function differs along the ith dimension*, [0116]), and

wherein in said evolutionary algorithm the number of marriages and of mutations of individuals are adaptive self-definable internal variables (*Suppose that any gene out*

of n possible genes, can get mutated with probability p, independently of other genes. In the Boolean setting, this is represented by a flip of value from 1 to 0 or vice versa and directly corresponds to the bit-flipping mutation operator in NK Landscapes, as well as in genetic algorithms and evolutionary computing. For Boolean networks, such random gene perturbations can be implemented with the popular DDLab software, [0124]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Shmulevich with the teachings of Agrafiotis for the purpose of modeling of complex systems, which include, but are not limited to gene regulatory networks, biological systems by utilizing methods that model the potential effect of individual genes on the global dynamical network behavior, both from the view of random gene mutation as well as intervention in order to elicit desired network behavior ([Abstract] of Shmulevich).

Agrafiotis, as modified by Shmulevich, does not disclose the mapping process from three-dimensional space to two-dimensional space is also used for atoms and molecules.

However, Blaney discloses the mapping process from three-dimensional space to two-dimensional space is also used for atoms and molecules (*Figure 1a is a representation of a protein, namely epidermal growth factor (EGF), showing its alpha carbon backbone B and the side chains TYR 13 (Tyrosine), CYS 14 (Cysteine), LEU 15 (Leucine), ASN 16 (Asparagine), and ARG 41 (Arginine), CYS 42 (Cysteine), and GLN 43 (Glutamine) of its putative active site AS. EGF has a complicated three-dimensional configuration, of which Figure 1a is a two-dimensional projection. Figure 1b is another*

projection of the molecule, rotated 90 degrees around the Y-axis which is shown for reference in each of these two figures, [Column 3, Lines 26-37]).

It would have been obvious to an ordinary person skilled in the art at the time of the invention was made to incorporate the teachings of Blaney with the teachings of Agrafiotis, as modified by Shmulevich, for the purpose of identifying the putative active site of a molecule, determining the structure of the active site, and, by means of a depth-first recursive procedure, generating designs of molecules which mimic that site to a desired degree of accuracy ([Column 1, Lines 58-64] of Blaney).

Regarding **claim 109**, Agrafiotis further discloses the evolutionary algorithm is a genetic algorithm (*The approach employs an iterative algorithm based on subset refinements to nonlinearly map a small random sample which reflects the overall structure of the data, and then "learns" the underlying nonlinear transform using a set of distributed neural networks, each specializing in a particular domain of the feature space*, [0045]).

Regarding **claim 110**, Agrafiotis further discloses:

a) for each atom a first and a second set of coordinates defining the position of the atom in the two or three dimensional space is calculated (*reducing the dimensionality of high-dimensional data in a way that preserves the original*

relationships of the data objects, and 2) producing Cartesian coordinate vectors from data supplied directly in the form of similarities or proximities, [0012]);

b) a fitness score of the matrix of distances among the atoms determined by means of the first and second set of coordinates defining the position of each atom in the two or three dimensional space is calculated by using the matrix of the known or measured distances between the atoms as a fitness function (*pairwise relationship and the current distance of the corresponding objects on the nonlinear map are illustrated in Figure 1, where shaded areas 110, 112 and 114 denote allowed ranges for a given pairwise relationship*, [0075]);

c) the first and second set of positional coordinates of each atom are combined according to predetermined combination rules, thus producing at least a new first and second set of positional coordinates for each atom (*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]);

d) a fitness score of said new first and second set of positional coordinates is calculated according to (b) (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]);

e) said new first and second set of positional coordinates are again combined according to (c) (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]); and

(c) through (e) are repeated until at least one new first or second set of positional coordinates reaches a maximum fitness score or is greater than a minimum predefined fitness score (*If d_{ij} is larger than r_{max}, the coordinates of the objects are updated using r_{max} as the target distance (Equation 9)*, [0071]).

Regarding **claim 111**, Agrafiotis further discloses for each combination of at least one first and one second set of coordinates several new sets of coordinates are obtained by combining said at least first and second set of coordinates(*Note that new patterns in R^n that not in the original input set can also be projected into R^m in the manner shown in Figure 6. Once the system is trained, new patterns in R^n are mapped by identifying the nearest local network and using that network in a feed-forward manner to perform the projection. The input for the system is a pattern 705 in R^n. This point is defined by its n attributes, x is $x_2, \ldots x_n$). The system includes a dispatcher module 710, which compares the distance of the input point to the network centers (i.e., the reference points), and forwards the input point to one of the available local neural networks 701, 702, or 703. Specifically, the input pattern is sent to the local neural network associated with the reference point nearest to the input pattern. The chosen network then performs the final projection, resulting in an output point in R^m*, [0110]).

Regarding **claim 112**, Agrafiotis further discloses at least one hidden or hypothetical atom is added to the database of geographical sites of which neither the coordinates nor the distances are known and a first and a second set of coordinates for said at least one atom are freely defined (*The general algorithm described above can also be applied when the pairwise similarity matrix is incomplete, i.e. when some of the pairwise similarities are unknown, when some of the pairwise similarities are uncertain or corrupt, or both of the above*, [0066]).

Regarding **claim 113**, Agrafiotis further discloses the calculation of the evolutionary algorithm is carried out in parallel for the database provided with and

without the at least one hidden or hypothetical atom (*multi-layer feed-forward networks trained to reproduce their inputs as desired outputs. They consist of an input and an output layer containing as many neurons as the number of input dimensions, and a series of hidden layers having a smaller number of units. In the first part of the network, each sample is reorganized, mixed, and compressed into a compact representation encoded by the middle layer. This representation is then decompressed by the second part of the network to reproduce the original input*, [0011]) and the best fit set of coordinates of position of the totality of the atoms obtained by the two parallel calculations are compared (*The points (objects) in each cell of the grid are then subjected to a phase of local refinement (i.e. only objects from within the same cell are compared and refined). Preferably, the number of sampling steps in each cell should be proportional to the number of objects contained in that cell. This process is highly parallelizable. This local refinement phase is then followed by another global refinement phase, and the process is repeated for a prescribed number of cycles, or until the embedding error is minimized within a prescribed tolerance*, [0086]).

Regarding **claim 114**, Agrafiotis further discloses a further pre-processing or post-processing phase is provided (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]).

Regarding **claim 115**, Agrafiotis further discloses in a preventive phase the known distance data matrix is subjected to treatment by means of a Self Organising Map algorithm (*The method uses a self-organizing principle to iteratively refine an initial (random or partially ordered) configuration of objects by analyzing only a subset of objects and their associated relationships at a time*, [0047]), the clusters formed by said SOM algorithm in the different units being then processed according to said method (*the reference points c_i are determined using a clustering algorithm described in greater detail below. In step 325, the training set T is partitioned into c disjoint clusters based on the distance of each point x_i from each reference point. The training phase concludes with step 335*, [0097]).

Regarding **claim 116**, Agrafiotis further discloses determine the presence and/or position of at least an unknown or hidden atom in the structure of the molecule (*The general algorithm described above can also be applied when the pairwise relationship matrix is incomplete, i.e. when some of the pairwise relationships are unknown. In this case, a similar algorithm to the one described above can be used, with the exception that the algorithm iterates over pairs of objects for which the relationships are known. In this case, the algorithm identifies configurations in space that satisfy the known pairwise relationships; the unknown pairwise relationships adapt during the course of refinement and eventually assume values that lead to a satisfactory embedding of the known relationships*, [0068]).

Conclusion

12. **THIS ACTION IS MADE FINAL**. Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Contact Information

13. Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Son T. Hoang whose telephone number is (571) 270-1752. The Examiner can normally be reached on Monday – Friday (7:00 AM – 4:00 PM).

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Neveen Abel-Jalil can be reached on (571) 272-4074. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/Son T Hoang/
Examiner, Art Unit 2165
July 28, 2009

/Neveen Abel-Jalil/
Supervisory Patent Examiner, Art Unit 2165